AD-A234 889

(2)

RADC-TR-90-404, Vol X (of 18)
Final Technical Report
December 1990

TIME ORIENTED PROBLEM SOLVING

Northeast Artificial Intelligence Consortium (NAIC)

James F. Allen

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

This effort was funded partially by the Laboratory Director's fund.

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

This report has been reviewed by the RADC Public Affairs Division (PA) and is releasable to the National Technical Information Services (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-90-404, Volume X (of 18) has been reviewed and is approved for publication.

APPROVED:

NORTHRUP FOWLER III
Project Engineer

APPROVED:

RAYMOND P. URTZ, JR.
Technical Director
Directorate of Command & Control

FOR THE COMMANDER:

BILLY G. OAKS
Directorate of Plans & Programs

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (COES) Griffiss AFB NY 13441-5700. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE December 1990	3. REPORT TYPE AND DATES COVERED Final Sep 84 - Dec 89

4. TITLE AND SUBTITLE
TIME ORIENTED PROBLEM SOLVING

5. FUNDING NUMBERS
C - F30602-85-C-0008
PE - 62702F
PR - 5581
TA - 27
WU - 13
(See reverse)

6. AUTHOR(S)
James F. Allen

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Northeast Artificial Intelligence Consortium (NAIC)
Science & Technology Center, Rm 2-296
111 College Place, Syracuse University
Syracuse NY 13244-4100

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Rome Air Development Center (COES)
Griffiss AFB NY 13441-5700

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
RADC-TR-90-404, Vol X (of 18)

11. SUPPLEMENTARY NOTES (See reverse)
RADC Project Engineer: Northrup Fowler III/COES/(315) 330-7794

This effort was funded partially by the Laboratory Director's fund.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The Northeast Artificial Intelligence Consortium (NAIC) was created by the Air Force Systems Command, Rome Air Development Center, and the Office of Scientific Research. Its purpose was to conduct pertinent research in artificial intelligence and to perform activities ancillary to this research. This report describes progress during the existence of the NAIC on the technical research tasks undertaken at the member universities. The topics covered in general are: versatile expert system for equipment maintenance, distributed AI for communications system control, automatic photointerpretation, time-oriented problem solving, speech understanding systems, knowledge base maintenance, hardware architectures for very large systems, knowledge-based reasoning and planning, and a knowledge acquisition, assistance, and explanation system.

The specific topic for this volume is a model theory and axiomatization of a logic for reasoning about planning in domains of concurrent actions.

14. SUBJECT TERMS Artificial Intelligence, Temporal Logic, Planning, Plan Recognition	15. NUMBER OF PAGES 28
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Block 5 (Cont'd) Funding Numbers

PE - 62702F	PE - 61102F	PE - 61102F	PE - 33126F	PE - 61101F
PR - 5581	PR - 2304	PR - 2304	PR - 2155	PR - LDFP
TA - 27	TA - J5	TA - J5	TA - 02	TA - 27
WU - 23	WU - 01	WU - 15	WU - 10	WU - 01

Block 11 (Cont'd)

This effort was performed as a subcontract by the University of Rochester to Syracuse University, Office of Sponsored Programs.

Contents

10.1 Executive Summary

The unifying theme underlying all of the research carried out under this project is that of producing new knowledge representation formalisms that extend the range of situations in which problem solving systems can be applied. These new formalisms have centered around issues related to time, temporal reasoning, and causation.

Significant progress has been made in the following areas:

- The axiomatic specification of an interval-based theory of time that allows for two different forms of time points ([AH87]);
- The development and public release of TIMELOGIC, an implementation in Common Lisp of the interval logic that quickly computes the relationship between arbitrary intervals using a constraint propagation algorithm;
- The development of a non-reified first-order temporal logic that has a well defined syntax, semantics, and proof-theory, and is easily implemented using a type-based theorem prover such as RHET ([BTK89]);
- The development of a logic that can represent and support reasoning about simultaneous interacting actions ([Pel88]);
- A generalized model of plan recognition, both as a formal theory and as a family of practical recognition algorithms ([Kau87]);
- The formal specification of two distinct forms of abstraction for planning systems, one based on reduced models, and the other an extension of inheritance hierarchies ([Ten88]);
- A new approach to causal reasoning that rejects overly-strong domain-independent approaches to solving the frame problem in favor of domain-dependent "cause-closure" axioms ([Web89]);
- A statistical/probabilistic approach to action reasoning that explicitly models that action success is not guaranteed, but subject to failure some proportion of the time ([Ten89]).
- The development and initial implementation of a problem solver that reasons, acts, and senses in real-time within the ARMTRAK model train domain;
- The development and public release of the HORNE representation system and its extension, RHET, two hybrid logic/frame-based representations for use in general problem solving and natural language understanding ([AM86, AM89]);

This work has been well received by the international artificial intelligence community, as evidenced by the publication of over two dozen articles and chapters in international journals and conferences. In addition, in recognition of their outstanding contributions to the field, two of the researchers on this grant received some of the highest honors that can

be given to computer scientists: James Allen, the principal investigator, recieved the Young Presidential Investigator's award, and Henry Kautz, (Ph.D. in 1987) received the Computers and Thought award.

10.2 Introduction

The overall aim of our research has been to develop planning representations that overcome the constraints and problems of earlier planning systems having impoverished temporal models and inadequate formal specifications. Much of our previous work has involved the development of representations that explicitly encode temporal knowledge, enabling one to reason about concurrent events [All84] and events beyond the agent's control [Pel88].

As a natural development of this research, we have additionally explored richer causal models and a novel approach to the frame problem, concurrent planning and activity, real-time problem solving, a formal account of plan recognition, abstraction in planning, and planning under uncertainty using probabilities. Much of our work is theoretical, since we believe that building robust, generalizable systems, in general, requires a mathematically rigorous framework. We have not, however, refrained from using implementations both as a testbed for new ideas, and to verify the strength of our theories. Therefore, our efforts at implementation occur concurrently with theory development. Implementations include

1. A plan recognizer, built on the formal theory of Henry Kautz, which finds the simplest explanation within a hierarchically structured plan library for a set of observations of atomic actions;

2. TIMELOGIC, an implementation of Allen's temporal logic that quickly computes the relationship between any pair of temporal intervals using an automatically generated interval reference hierarchy;

3. The TEMPOS reasoning system, which integrates TIMELOGIC and RHET, together with supporting predicates for interval-based recurrence logic;

4. ARMTRAK, a simulator for a model train domain;

5. A prototype of a real-time planning system for the model train domain, that interfaces not only with the ARMTRAK simulator, but also with a physical model train set having software settable switches, and which can be perceived by an active vision system [Bal89];

6. Two general knowledge representation systems, HORNE and RHET; which provide powerful inference capabilities, and additionally have well defined semantics. Both have been publicly distributed.

The results of this effort to extend planning formalizations beyond that of standard approaches have been impressive, as evidenced by the frequent publication of this work in international journals and conferences, and the awards and honors given to our researchers. In the following sections, we will describe these research efforts in more detail.

10.3 Time-Oriented Problem Solving

10.3.1 Time Points and Moments

Interval-based temporal reasoning, since its introduction by the PI in 1983, has proven to be a useful formalism in many different areas. In the original work, a taxonomy of thirteen binary relations was defined that completely characterize the relationships that can hold between any two temporal intervals. These include the six relations: MEETS, OVERLAPS, STARTS, FINISHES, DURING, BEFORE, as well as their inverses (i.e., switching the order of the arguments, since all are asymmetric), and the relation EQUAL. In addition, a constraint propagation algorithm was developed that allowed for the efficient querying and updating of a knowledge base of temporal assertions.

This work was extended under this project to include a concise formal theory for the interval logic and to investigate the relationship between interval and point based temporal representations. In conjunction with Pat Hayes[1], the new theory developed had the following features:

- The introduction of time periods as the basic building block, which does not commit to either being an interval or a (certain form of) point;
- A concise axiomitization (five axioms) that completely captures the inferential capability of the interval logic;
- The definition of all 13 binary temporal relationship with only a single primitive predicate, MEETS;
- The development of two distinct notions of point-like time periods
  1. *Points*, which are endpoints of periods and can be shown not to be periods themselves but are useful for representing times of transition,
  2. *Moments*, which are non-decomposable periods, and are useful for representing "instantaneous" events, such as flashes, clicks, etc.
- Demonstrating that this logic is consistent with various models of discrete time, continuous time, or models that mix the two.

10.3.2 TIMELOGIC

TIMELOGIC is an interval-based forward-chaining inference engine and database manager of temporal constraints. Relational constraints indicating relative temporal order between intervals is maintained, based on Allen's interval logic. Partial temporal knowledge can be explicitly maintained, from which all possible consistent temporal orderings can be inferred. This is done by maintaining sets of possible temporal relations between any *two* intervals, and providing a temporal constraint propagation algorithm. Knowledge can be added to a network of interval constraints by reducing the set of possible temporal relations between

[1]Now at Xerox PARC

10.3 Time-Oriented Problem Solving

10.3.1 Time Points and Moments

Interval-based temporal reasoning, since its introduction by the PI in 1983, has proven to be a useful formalism in many different areas. In the original work, a taxonomy of thirteen binary relations was defined that completely characterize the relationships that can hold between any two temporal intervals. These include the six relations: MEETS, OVERLAPS, STARTS, FINISHES, DURING, BEFORE, as well as their inverses (i.e., switching the order of the arguments, since all are asymmetric), and the relation EQUAL. In addition, a constraint propagation algorithm was developed that allowed for the efficient querying and updating of a knowledge base of temporal assertions.

This work was extended under this project to include a concise formal theory for the interval logic and to investigate the relationship between interval and point based temporal representations. In conjunction with Pat Hayes[1], the new theory developed had the following features:

- The introduction of time periods as the basic building block, which does not commit to either being an interval or a (certain form of) point;
- A concise axiomitization (five axioms) that completely captures the inferential capability of the interval logic;
- The definition of all 13 binary temporal relationship with only a single primitive predicate, MEETS;
- The development of two distinct notions of point-like time periods
    1. *Points*, which are endpoints of periods and can be shown not to be periods themselves but are useful for representing times of transition,
    2. *Moments*, which are non-decomposable periods, and are useful for representing "instantaneous" events, such as flashes, clicks, etc.
- Demonstrating that this logic is consistent with various models of discrete time, continuous time, or models that mix the two.

10.3.2 TIMELOGIC

TIMELOGIC is an interval-based forward-chaining inference engine and database manager of temporal constraints. Relational constraints indicating relative temporal order between intervals is maintained, based on Allen's interval logic. Partial temporal knowledge can be explicitly maintained, from which all possible consistent temporal orderings can be inferred. This is done by maintaining sets of possible temporal relations between any *two* intervals, and providing a temporal constraint propagation algorithm. Knowledge can be added to a network of interval constraints by reducing the set of possible temporal relations between

[1] Now at Xerox PARC

syntax and proof theory means that we can take advantage of over 20 years of research in automated reasoning. Since our logic is first-order, we can avail ourselves of the current automated reasoning technology.

We show that Shoham's logic is subsumed by the logic we have developed by defining two transformations, a syntactic transformation, π_{syn}, and a semantic transformation, π_{sem}. π_{syn} maps sentences of STL to sentences of BTK, while π_{sem} maps models of STL to models of BTK. Using these two transformations we can show that any STL model is transformable into a BTK model in such a way that the set of sentences satisfied by the BTK model[4] includes the transformed set of STL sentences satisfied by the STL model. In other words, any set of STL sentences can be rewritten as a set of BTK sentences without eliminating any models which satisfy those sentences.

10.3.4 Concurrent Actions and External Events

Allen's temporal logic can be used to describe what actually happens over time, by the assertion of propositions that are associated with the temporal interval in which they are true. However, it cannot be used to describe different possibilities that a planning agent might be able to bring about. In particular, there is no ability for the agent to do counterfactual reasoning about possible futures, which the *result* function of *situation-calculus* [MH69] permits. To accommodate this counterfactual reasoning, Pelavin's Ph.D dissertation [Pel88] extended Allen's logic with two modal operators. The first operator captures temporal possibility, allowing the distinction to be made between propositions that are possibly true, and those that are inevitably true. The second operator, *IFTRIED*, captures external events, and allows the distinction to be made between those events that are under the agent's control and those that are not.

A formal specification of this logic is provided, based upon the *possible worlds semantics* developed by Hintikka [Hin62] and Kripke [Kri63]. Possible worlds in Pelavin's logic refer to *world-histories*, that is, complete characterizations of worlds over time. Each sentence is thus given a truth value relative to a world history. The treatment of the *IFTRIED* operator is derived from the semantic theories of conditionals developed by Stalnaker [Sta85] and Lewis [Lew73].

A proof theory was then developed that is sound with respect to the semantics. The axiomitization of the interval logic fragment is a syntactic variation of Allen and Hayes's [AH87]. The inevitability modal operator behaves like the $S5$ necessity operator for a fixed time argument, with an axiomitization taken from Hughes and Cresswell [HC68]. The axioms and rules capturing *IFTRIED* can be divided into three categories: properties relating to a subjunctive conditional, the relation between *IFTRIED* and the inevitability operator, and properties describing the effect of attempting plan instance composition.

Finally, a non-linear planning algorithm was developed that exploits axioms stating the *non-interference* relations between different plan instances. This algorithm differs from previous planning systems in the method used to handle action interactions and the use of explicit plan instances to *maintain* properties over temporal intervals. The interaction of

[4] A model, $\mathcal{M}$, satisfies a sentence, α, written $\mathcal{M} \models \alpha$, if $\alpha^\sigma = \top$, i.e., if α is true under σ, the interpretation of the model.

two or more concurrent or sequential plan instances will be computed by considering the interaction of two plan instances that overlap in time. The frame problem is thus handled by determining the conditions under which concurrent plan instances interfere with each other. This therefore obviates the need to use the STRIPS assumption or the persistence assumption to describe what actions do not affect.

10.3.5 A Formal Theory of Plan Recognition

Researchers in discourse analysis, story understanding, and user modeling for expert systems have shown great interest in plan recognition problems. In a plan recognition problem, one is given a fragmented description of actions performed by one or more agents, and expected to infer the overall plan or scenario which explains those actions. This project, forming the core of Henry Kautz's Ph.D. dissertation, [Kau87] developed the first formal description of the plan recognition process.

Beginning with a reified logic of events, a scheme for structuring a plan library was devised, based on both a taxonomic and a partonomic hierarchy. A semantic basis for non-deductive inference, called *minimum covering entailment*, justifies the conclusions that one may draw from a set of observed actions. Minimum covering entailment is defined by delineating the class of models in which the library is complete and the set of unrelated observations is minimized. An equivalent proof theory forms a preliminary basis for mechanizing the theory. Equivalence theorems between the proof and model theories is presented. Minimum covering entailment is related to a formalism for non-monotonic inference known as *circumscription*. Finally, the thesis describes a number of algorithms which correctly implements the theory, together with a discussion of their complexity.

The theory has been applied to a number of examples of plan recognition, in domains ranging from an operating system advisor to the theory of speech acts. The thesis shows how problems of medical diagnosis, a similar kind of non-deductive reasoning, can be cast in the framework, and an example previously solved by a medical expert system is worked out in detail. It has also been ported to BBN Labs for use in their FRESH/OSGP (Naval Search and Rescue) domain.

The analysis provides a firm theoretical foundation for much of what is loosely called *frame based inference*, and directly accounts for problems of ambiguity, abstraction, and complex temporal interactions, which were ignored by previous work. The framework can be extended to handle difficult phenomena such as error, and can also be restricted in order to improve its computational properties in specialized domains.

10.3.6 Abstraction in Planning

Planning involves reasoning about the effect of actions on the world. Josh Tenenberg's dissertation [Ten88] explores the use of abstraction in planning in order to simplify the reasoning task for an automated agent.

Abstraction has been explored in two settings. In the first, relaxing the operator preconditions of a STRIPS system [FN71] was used in a manner similar to that of Sacerdoti in his ABSTRIPS system [Sac74]. Tenenberg attacked the problem that plagues Sacerdoti's work, of the potential for inconsistencies to arise at the abstract levels, by assigning criticalities to

equivalence classes of preconditions, where two preconditions belong to the same class if they occur together in the same domain axior This type of abstraction was shown to preserve consistency. In addition, it was demonstrated that for every concrete level plan P solving some problem, there exists at the abstract level a plan P' which also solves the problem, where P' includes at most a subset of the $P's$ plan steps in the same relative order.

In the second setting, a generalization of inheritance hierarchies is used as the basis for abstraction. Inheritance is applied not only to object classes but also to relations between objects, and actions on objects. A syntactic mapping between theories is presented that rigorously describes how a concrete level theory can be mapped to a. abstraction, such that the abstraction preserves the desired inheritance semantics. In addition, there is a strong proof-theoretic property that states that for any plan that provably solves a problem at the abstract level, there exists a plan at the concrete level with a similar structure that is guaranteed to solve some specialization of the abstract problem. This property provides a very strong constraint on search for those problems for which abstract solutions exist. This type of abstraction additionally holds promise for attacking such issues as saving and reusing plans (rather than always constructing plans from first principles), and planning with disjunctive uncertainties.

10.3.7 Domain-Dependent Causal Reasoning

Historically, the *frame problem* has been discussed within the context of the situation calculus [MH69]. It arises, however, in representations incorporating an explicit discrete time line, where it is typically called the *persistence problem.* Unfortunately, promising solutions from situation calculus did not transfer to these representations.

Several researchers have independently suggested solving this problem using circumscription so that explicit *frame axioms* stating which propositions *do not* change truth value as a result of an action are not required. Most recently, these proposals suggest circumscribing a causality predicate, so that models are preferred in which the only propositions changing truth value are those for which a cause is explicitly known. However, there are serious problems with these solutions, in particular, the *generation/minimization problem* [Web89]. This arises when the agent is able to infer (non-monotonically) from its ignorance about whether a particular action occurred, to the fact that the action in fact did *not* occur. One is thus using a domain-independent rule that sanctions the inference of domain facts purely, (and spuriously) from ignorance.

Fueled by these objections, Jay Weber, in his Ph.D. dissertation [Web89], has developed a formal theory of action reasoning based on an explicit time line that has the following features:

- The causality predicate is not circumscribed. Instead, non-occurrences are entailed by domain-dependent *executability axioms* which specify *necessary* conditions for action occurrences. An action is believed to not occur when at least one of its necessary conditions is believed to not hold.

- Effects of actions are entailed by axioms of the form:

$$\forall m[\bigvee_{i=1}^{\nu} \mathtt{occurs}(\epsilon_i, m) \rightarrow \mathtt{holds}(\phi, m)]$$

When adding the definition of negation, this form extends to negative effects:

$$\forall p, t[\texttt{holds}(p,t) \equiv \neg\texttt{holds}(\overline{p},t)].$$

- Non-effects of actions are entailed by *cause closure axioms* [Sch89], which specify the possible conditions for property change, as in:

$$\forall m[(\texttt{holds}(\phi,m) \land \neg\texttt{holds}(\phi,m')) \rightarrow \bigvee_{i=1}^{\nu} \texttt{occurs}(\epsilon_i,m)]$$

  By the contrapositive, a property is believed to remain holding when it is believed that no possible cause occurs.

- External events (not inspired by the planning agent) are supported through *event enabling axioms*, which specify sufficient conditions for action occurrences. This ability to infer the occurrence of events based solely on properties is actually quite novel, and because of this, past formal frameworks have not supported external events.

The above features constitute a solution to the persistence problem which does not suffer from the generation/minimization problem, nor does it create an ambiguity between action terms. For example, the theory demonstrates how an explicit time version of the shooting problem scenario does not contain enough information to produce the popular interpretation of the outcome in a domain-independent manner.

10.3.8 Probabilistic Action Reasoning

Previous approaches to action reasoning have been dominated by approaches that either assume the completeness of the planning domain, as in STRIPS [FN71], or make overly strong assumptions that the agent's knowledge is complete. Although the domain-dependent approaches do not suffer from these problems, they are inherently weak, in that they do not sanction the number of inferences that are often required. That is, they only allow one to infer that a proposition persists if it can be proven that none of the actions that cause the proposition to change truth value have occurred. As an alternative, Weber and Tenenberg [Web89, TW89, Ten89] have been exploring the use of probabilistic information as a better model of uncertainty in action reasoning.

A close relative of the frame and persistence problems is the *qualification* problem. This problem concerns what preconditions an agent considers sufficient for an action to achieve an effect. In general, the consideration of an ideal list of sufficient preconditions will be impossible or impractical, and as such, an agent reasoning about action success will be obliged to do so from incomplete evidence. Standard approaches to this problem have been to use non-monotonic or consistency based logical methods that assume those sufficient preconditions which are usually true are true by default. However, these approaches all suffer from a classic problem of default logics called the *lottery paradox*, as a result of the coarse way that defaults capture statistical properties of the domain.

In contrast, Weber and Tenenberg developed a novel method for solving the qualification problem using a form of statistical inference developed by Henry Kyburg [Kyb74]. The

agent acquires statistics about the proportion of success of its actions, conditioned upon the existence of certain preconditions which hold just prior to the action. For instance, the agent has sentences that express "the proportion of worlds in which the car starts, given that the key is turned, and there is gas in the car is in the interval [.8,.9]." From such statistics, the agent derives degrees of belief in the success of *particular* actions. That is, such statistics (under certain formal conditions) sanction the agent to associate a probability with the sentence "the car will start at Time37."

Choosing among a set of applicable statistics is the familiar *reference class* problem, which is the subject of much work on statistical inference. It is demonstrated that each qualification serves to define a more specific reference class, and that although no completely specified reference class will be obtainable, more general classes are nonetheless useful because they capture statistical generalizations about the omitted qualifications. Since this approach quantitatively captures the statistical properties of the domain, it does not suffer from the lottery paradox as do default approaches. Lastly, it is shown that this approach also provides a rational basis for solving the frame (persistence) problem. In fact, the solution to the frame and qualification problem use the same representation: statistical assertions relating the likelihood that a proposition will have a truth value in the resulting situation, conditioned upon the occurrence of accompanying actions, and the truth of particular preconditions.

10.3.9 ARMTRAK

Research most recently has been marked by continued development of the ARMTRAK planning domain and the development of a planner capable of generating plans which recognizes the uncertainty of the domain. ARMTRAK captures the conceptual simplicity of the blocks world while extending it sufficiently to exercise planners with more realistic capabilities. Several graduate student researchers, most notably Steven Feist and Nathanial Martin, have been working with James Allen and Josh Tenenberg, in the development and implementation of ARMTRAK.

The inputs to the ARMTRAK are simple: switch settings and engine power; but many different scenarios can be generated from these simple inputs. Because the train's movement is independent of the planner, plans which succeed at one point may fail as the execution progresses. The planner must therefore take into account temporal constraints on the generation and execution of its plans. Moreover, model trains are notorious for falling off the track. Therefore, plans may succeed or fail due to unexpected effects of the primitives. Finally, in certain cases there may be no plans which achieve the desired goal, but which may partially achieve that goal. In this case, all possible plans fail, and the planner must select the most useful plan from the set of failing plans.

Moreover, the information necessary for the domain is supplied by three predicates: switch-setting, train-location, and train-speed. The three predicates necessary to use the Rochester Robot to plan in the ARMTRAK domain have been finished. The robot is now able to discern when a train is present and what the states of the switches are. A programmer's manual for the current situation is in preparation [Mar89].

The planning domain, implemented by Nat Martin, is an extension of the ARMTRAK domain as the planner is forced to consider the position of the robot before it is able to tell the location and speed of the trains and the states of the switches. This partial implementation

will, however, provide a basis for implementing the ARMTRAK domain. The predicates as they are defined in ARMTRAK will be implemented as reactive behaviors which the planner can execute. Such an implementation is reminiscent of Shakey, the Stanford mobile robot. Shakey's planning predicates were intermediate level actions (ILA's) written in a language called Markov based on low level actions (LLA's) written in a language that could be compiled directly into instructions for the hardware.

The implementation of the domain was simplified so that experimentation could begin before the domain was completed. Instead of providing the three queries and two commands, three programs were written. The first program determines whether or not the switch is closed. The second performs a df/dt on the images streaming through the camera to find the trains by their movement. The third will calculate the speed of the train from the change in the center of gravity of the moving objects found by the second program. The commands are simulated by sending requests to the user to change the setting of the switches and the power supply of the trains. The controls for the train have been fed through the wall in the robot room so the user can operate the trains.

10.3.10 The HORNE and RHET Representation Languages

In 1986, the implementation of the HORNE reasoning system was completed. This system is a hybrid logic/frame based representation for use in general problem solving and natural language understanding. HORNE has been distributed to over 50 research institutions throughout the lifetime of the project.

Based on the experiences gained from implementing HORNE, the RHET system was developed, adding several important extensions to HORNE, and making significant improvements with respect to software engineering. RHET offers a set of formally well defined tools for building an automated reasoning system. It's emphasis is on flexibility of representation, allowing the user to decide if the system will basically operate as a theorem prover (e.g. like PROLOG), a frame-like system (e.g. like KL-1), or an associative network.

RHET offers two main modes of inference: a horn clause theorem prover, and a forward chaining mechanism. To supplement these modes, RHET includes specialized reasoners for dealing with equality of ground terms, types of terms (any of ground terms, variables, or functions), contextual reasoning, and constraint proof mechanisms. RHET allows a reasonably functional interface with Lisp allowing one to escape from RHET to Lisp, or from Lisp to RHET as needed[5].

The full RHET language is being supported and is running in an interpreted fashion. In particular, the following has been achieved:

- Publication of two TRs, numbers 238 and 239 in February 1988, and revised in October of 88. These are the User Manual and Programmer's Guide respectively.

- Structured type reasoning is fully implemented with the exception of certain advanced functionality, e.g. classification[6].

[5]Normally, one would either code one's project in Lisp and escape to RHET to handle the KR tasks, or code in RHET and possibly escape into Lisp for efficiency or special i/o issues.

[6]Exceptions to the full implementation of the RHET language are carefully noted in TR238.

- An inequality reasoner has been added, which keeps the prover from assuming or allowing two objects that have been declared unequal to be made equal.

- Belief modal operators are fully supported.

- Structured type definitions have been extended to include uninterpreted relations, which allow the user to associate arbitrary RHET objects with a type declaration. For example, one might define a *Precondition* relation on a plan type whose value would be a list of preconditions for executing a plan instance of this type.

10.4 Contributions to the Scientific Community

In addition to the research carried out under this grant, the researchers have made contributions to the artificial intelligence community, through their continued work in the leading research labs and academic institutions upon graduating from the University of Rochester, through publications in journals, conferences, and books, and through the hosting of a Planning Workshop in 1988 bringing together many of the world's finest planning researchers. In addition, individuals on this project received some of highest distinctions awarded to researchers in artificial intelligence.

Throughout its duration, this grant has supported a number of researchers: upperclass undergraduates (2, helping with software development), graduate students (9 total, all receiving M.S. degrees and 4 receiving Ph.D's), staff programmers(4), and research associates(3). Those graduating with Ph.D's are working at Philips Labs, AT&T Bell Labs, Lockheed, and at the University of Rochester.

During the granting period, funded researchers have contributed over two dozen articles at international conferences/workshops and journals. In several of these conferences, where even a single paper by a graduate student is exceptional, our graduate students have had multiple papers accepted, with Josh Tenenberg having two papers apiece in IJCAI-87 and KR-89, and Henry Kautz having three papers in AAAI-86. In addition, James Allen has published a book on Natural Language processing, which has been quickly adopted as the most comprehensive and scholarly text in the field. In addition, he has contributed 12 articles/chapters in books during this time. In progress is a book co-authored by James Allen, Henry Kautz, Richard Pelavin, and Josh Tenenberg, that provides a detailed overview of the formal planning research carried out by these researchers, in temporal planning, plan recognition, and plan abstraction.

An international workshop on Planning in AI was organized by James Allen, Josh Tenenberg, and Jay Weber, and hosted by the department in Fall, 1988. The workshop was supported jointly by the Computer Science Department of the University of Rochester, AAAI, and NAIC. The thrust of the workshop was to explore the scientific and engineering issues that currently impede progress in the development of systems that solve real planning problems in real environments. Our interest in these issues was not limited to robotic applications, but included a broad range of others, including natural language understanding, scheduling, and distributed problem solving. Approximately sixty people attended the workshop, all of whom are actively engaged in planning research. A number of the presentations offered are quite novel, and deserving of the publication of the proceedings [TWA89] whould foster a

wider dissemination of this work within the planning research community. We believe that workshops of the kind that we hosted are essential for the vitality of artificial intelligence as a field. Our institution, is committed to continuing this kind of interchange with the larger research community.

We are especially proud of one of our recent NAIC graduates, Henry Kautz, who has just been awarded the "Computers and Thought" award, one of the most prestigious distinctions in the field. This award is given every two years by an international committee of scholars, who choose a researcher who has made significant contributions to artificial intelligence. His dissertation, completed in 1987 under James Allen, provided the first formal accounts of plan recognition.

In addition, James Allen was the recipient during this period of a National Science Foundation Presidential Young Investigator Award. Only a small handfull of such awards are presented each year, and competition for these awards is not just among AI researchers, but among researchers in all scientific disciplines.

10.5 Publication List

Allen, J.F., "A General Model of Action and Time," *Artificial Intelligence*, 1985.

Allen, J.F., "Speech Acts," in *Encyclopedia of Artificial Intelligence*, Shapiro, S. (ed.), John Wiley & Sons, 1986.

Allen, J.F., *Natural Language Understanding*, Benjamin Cummings Publishing Co., Inc., 1987.

Allen, J.F., Giuliano, M., and Frisch, A., "The Horne Reasoning System," TR 126 (revised), Department of Computer Science, University of Rochester, 1984.

Allen, J.F. and Hayes, P.J., "A Commonsense Theory of Time," IJCAI, Los Angeles, CA., 1985.

Allen, J.F. and Hayes, P.J., "Moments and Points in an Interval-Based Temporal Logic," (submitted December 1988), Computational Intelligence.

Allen, J.F. (with Hayes, P.J.) "Short Time Periods," IJCAI, Milan, Italy, 1987.

Allen, J.F. and Kautz, H., "A Model of Naive Temporal Reasoning," in *Formal Theories of the Commonsense World*, Hobbs, J. and Moore, R. (eds.), Ablex Publishing Co., 1985.

Allen, J.F., (with Litman, D.) "A Plan Recognition Model for Subdialogues in Conversations," *Cognitive Science 11:2*, 1987.

Allen, J.F. and Miller, B.W., "The Rhetorical Knowledge Representation System: A User's Manual," TR 238, Computer Science Department, University of Rochester, revised 1989.

Guez, S., "Even Argumentation is Linguistic," (submitted December 1988), *International Joint Conference on Artificial Intelligence.*

Hinkelman, E., and Allen, J.F., "Two Constraints on Speech Act Ambiguity," ACL, Vancouver, B.C., 1989.

Kautz, H., "Toward a Theory of Plan Recognition," TR 162, Computer Science Department, University of Rochester, 1985.

Kautz, H., "The Logic of Persistence," AAAI, Philadelphia, PA., 1986.

Kautz, H. and Allen, J.F., "Generalized Plan Recognition," AAAI, Philadelphia, PA., 1986.

Kautz, H., (with Vilain, M.) "Constraint Propagation Algorithms for Temporal Reasoning," AAAI, Philadelphia, PA., 1986.

Miller, B.W., "The Rhet Programmer's Guide (for Version 14.45)," TR 239, Computer Science Department, University of Rochester, revised 1989.

Pelavin, R., and Allen, J.F., "A Model of Concurrent Actions Having Temporal Extent," AAAI, Seattle, WA., 1987.

Porat, S., "Fairness in Context-Free Grammars under Every Choice of Strategy," *Information and Control*, 1988.

Tenenberg, J.D., "Taxonomic Reasoning," IJCAI, Los Angeles, CA., 1985.

Tenenberg, J.D., "Planning with Abstraction," AAAI, Philadelphia, PA., 1986.

Tenenberg, J.D., "Preserving Consistency across Abstraction Mappings," IJCAI, Milan, Italy, 1987.

Tenenberg, J.D., "Abstraction in Planning," Ph.D. Dissertation and TR 250, Computer Science Department, University of Rochester, May, 1988.

Tenenberg, J.D., "Extending Inheritance Abstraction to Symbolic Planning Systems," 1988-89 *Computer Science and Engineering Research Review*, University of Rochester, October, 1988.

Tenenberg, J.D., "Inheritance in Automated Planning," *First International Conference on Principles of Knowledge Representation and Reasoning*, Toronto, Canada, 1989.

Tenenberg, J.D., "The Robot Designer's Dilemma," *First International Workshop on Human and Machine Cognition*, Pensacola, Florida, 1989.

Tenenberg, J.D., (with Fahiem Bacchus and Hans Koomen), "A Non-Reified Temporal Logic," *First International Conference on Principles of Knowledge Representation and Reasoning*, Toronto, Canada, 1989

Tenenberg, J.D., (with Leo Harman), "Performance in Practical Problem Solving," IJCAI, Milan, Italy, 1987.

Tenenberg, J.D., Weber, J., and Allen, J.F. (Eds.), "Proceedings from the Rochester Planning Workshop: From Formal systems to Practical Systems," TR 284. Computer Science Department, University of Rochester, 1989.

Weber, J.C., "Plan-Based Goal-Oriented Classification," TR 243, Computer Science Department, University of Rochester, February, 1988.

Weber, J.C., "A Versatile Approach to Action Reasoning," TR 237, Computer Science Department, University of Rochester, March, 1988.

Weber, J.C., "The Myth of Domain Independent Causal Reasoning," *First International Workshop on Human and Machine Cognition*, Pensacola, Florida, 1989.

Weber, J.C., "A parallel Algorithm for Statistical Belief Refinement and Its Use in Causal Reasoning," IJCAI, Detroit, MI, 1989.

Weber, J.C., "Principles and Algorithms for Causal Reasoning with Uncertain Knowledge," Ph.D. Dissertation, TR 287, Computer Science Department, University of Rochester, February, 1988.

Whitehead, S. and Ballard, D., "A Role for Anticipation in Reactive Systems that Learn," *Sixth International Workshop on Machine Learning*, Ithaca, NY, 1989.

Whitehead, S. and Ballard, D., "Reactive Behavior, Learning, and Anticipation," NASA Conference on Space Telerobotics, Pasadena, CA, 1989.

Bibliography

[TWA89] Tenenberg, J.D., Weber, J., and Allen, J.F. (Eds.). Proceedings from the Rochester Planning Workshop: From Formal systems to Practical Systems, TR 284, Computer Science Department, University of Rochester, 1989.

[Bal89] Dana Ballard Animate Vision. IJCAI, Detroit, MI, 1989.

[All84] James Allen. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM* 26:11, 1983.

[Mar89] Nathanial Martin. Armtrak Programmer's Manual (in preparation).

[HC68] G. Hughes and M. Cresswell. *An Introduction to Modal Logic*, Methuen and Co., Ltd., 1968.

[Lew73] Lewis, D. *Counterfactuals*, Harvard University Press, Cambridge, MA, 1973.

[Sta85] Stalnaker, R. *Inquiry*, MIT Press, Cambridge, MA, 1985.

[Kri63] Kripke, S. Semantical Analysis of Mod-l Logic. Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik, 9, 67-96, 1963.

[Hin62] Hintikka, J. *Knowledge and Belief: An Introduction to the Logic of Two Notions*, Cornell University Press, 1962.

[MH69] John McCarthy and Patrick J. Hayes. Some Philosophical Problems from the Standpoint of Artificial Intelligence. In Bonnie Lynn Webber and Nils J. Nilsson, editors, *Readings in Artificial Intelligence*, 432–450, Tioga, 1981.

[AM86] James Allen and Bradford Miller. The HORNE Reasoning System in COMMON LISP. TR 126 (revised), University of Rochester, 1986.

[AM89] James Allen and Bradford Miller. The Rhetorical Representation System: A User's Manual. TR 238 (rerevised), University of Rochester, 1989.

[BTK89] Fahiem Bacchus, Josh Tenenberg and Hans Koomen. A Non-Reified Temporal Logic. First Internation Conference on Principles of Knowledge Representation and Reasoning, Toronto, Ontario, Canada, 1989.

[Web89] Jay Weber. Principles and Algorithms for Causal Reasoning with Uncertainty. Ph.D. Dissertation and TR 287, University of Rochester, 1989.

[Kau87] Henry Kautz. A Formal Theory of Plan Recognition. Ph.D. Dissertation and TR 215, University of Rochester, 1987.

[Pel88] Richard Pelavin. A Formal Approach to Planning with Concurrent Actions and External Events. Ph.D. Dissertation and TR 254, University of Rochester, 1988.

[Ten88] Josh Tenenberg. Abstraction in Planning. Ph.D. Dissertation and TR 250, University of Rochester, 1988.

[Ten89] Josh Tenenberg. The Robot Designer's Dilemma. The First International Workshop on Human and Machine Cognition, Pensacola, Florida, 1989.

[AH87] James Allen and Patrick Hayes. Moments and Points in an Interval-Based Temporal Logic. TR 180, University of Rochester, 1987.

[FN71] Richard Fikes and Nils Nilsson. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence* 2:189-208, 1971.

[Sac74] Earl Sacerdoti. Planning in a Hierarchy of Abstraction Spaces. *Artificial Intelligence* 5, 1974.

[Sch89] Lenhart Schubert. Solving the original frame problem without frame axioms or nonmonotonicity. In Henry Kyburg and Ron Loui, editors, *Selected papers from the 1988 Society for Exact Philosophy Conference*, 1989.

[Sho88] Yoav Shoham. *Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence.* The MIT Press, Cambridge, MA, 1988.

[Wa52] Hao Wang. Logic of Many-Sorted Theories. *JSL* 17, 1952.

[Wal87] Christoph Walther. A Many-Sorted Calculus Based on Resolution and Paramodulation. Pitman Research Notes in Artificial Intelligence 1987.

[TW89] Josh Tenenberg and Jay Weber. A Statistical Solution to the Qualification Problem and How it Also Solves the Frame Problem. /newblock Submitted to the First International Workshop on Human and Machine Cognition, Pensacola, Florida, 1989.

[Kyb74] Henry Kyburg, Jr. *The Logical Foundations of Statistical Inference*, Reidel, 1974.

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control, Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of competence is provided to ESD Program Offices (POs) and other ESD elements to perform effective acquisition of C^3I systems. The areas of technical competence include communications, command and control, battle management information processing, surveillance sensors, intelligence data collection and handling, solid state sciences, electromagnetics, and propagation, and electronic reliability/maintainability and compatibility.